ESTUDO E IMPLEMENTAÇÃO DE ALGORITMOS PARA RESOLUÇÃO DE PROBLEMAS COMBINATÓRIOS EM GRAFOS

TRABALHO DE INICIAÇÃO CIENTÍFICA

Aluno da Universidade de Taubaté – Bolsa PIBIC/CNPq

ORIENTADOR

Dr. Horacio Hideki Yanasse, Pesquisador Titular, LAC – INPE

Fevereiro de 1.999 a Fevereiro 2.000

Relatório de Atividades

Título do Projeto: ESTUDO E IMPLEMENTAÇÃO DE ALGORITMOS PARA RESOLUÇÃO DE PROBLEMAS COMBINATÓRIOS EM GRAFOS

Bolsista: Ralphy Antonio Martin Castilho

Orientador: Dr. Horacio Hideki Yanasse

Período: Agosto de 1999 a Fevereiro de 2000

ÍNDICE

INTRODUÇÃO

A questão mais comum que vem à cabeça da maioria dos motoristas que partem de um local à outro é a escolha de uma rota a ser seguida. Dentre as alternativas de rota a serem avaliadas, uma tem de ser escolhida com base nos critérios como distância percorrida, o tempo da viagem, custo do combustível e confiabilidade da rota. Problemas deste tipo são parte de uma ampla gama de outros problemas denominados de Problemas de Roteamento de Veículos (PRV). Vários problemas de roteamento podem ser definidos através de uma entidade abstrata conhecida como "grafo".

Um grafo é um conjunto de arcos (ou arestas) e nós (ou vértices), sendo que cada arco conecta um nó a si mesmo ou um par de nós (um em cada extremidade do arco). O grafo pode representar por exemplo, um mapa rodoviário de uma cidade, onde os arcos e os nós corresponderiam às ruas e aos cruzamentos, respectivamente.

Vários PRV já foram estudados e algoritmos especializados para sua solução estão disponíveis na literatura. Pode-se destacar, por exemplo, o algoritmo de Dijkstra, que obtém o caminho mínimo de um determinado nó à outro, e o algoritmo de Floyd, que obtém o caminho mínimo entre todos os pares de nós do grafo. Destacam-se também outros algoritmos mais complexos, para resolução do Problema do Caixeiro Viajante e do Problema do Carteiro Chinês. Estes problemas são clássicos dentro do PRV e possuem formas de solução bem conhecidas. (Vide, por exemplo, Christofides, 1975).

O problema do Caixeiro Viajante (PCV) tem merecido uma grande atenção por parte dos pesquisadores por auxiliar na solução de diversos problemas de seqüenciamento de atividades. Esse problema consiste na determinação da rota de menor custo para uma pessoa que parta de uma cidade e deva visitar diversas outras, retornando depois à cidade de origem.

No Problema do Carteiro Chinês (CPP) a finalidade é partindo de um determinado nó do grafo, percorrer todos os arcos deste grafo pelo menos uma vez, e que este percurso tenha a distância mínima possível.

OBJETIVO

O objetivo principal deste projeto é o estudo e implementação de algoritmos para resolução de problemas combinatórios em grafos. No período foram estudados os algoritmos do Caminho Mínimo de Dijkstra e o de Floyd, o Problema do Caixeiro Viajante (PCV), do Carteiro Chinês (CPP), descritos em Sampaio (1998) e também implementados pelo mesmo autor.

Atualmente estão sendo objeto de estudo um algoritmo para Localização de Facilidades e outro para resolução do problema do Carteiro Chinês para grafos orientados. O algoritmo para Localização de Facilidades consiste em determinar boas localizações para a construção de pontos de facilidades ou serviços públicos. O algoritmo do Carteiro Chinês em grafos orientados percorre todos os arcos do grafo respeitando a restrição do sentido (mão única) dos arcos, o que geralmente dificulta mais este problema de roteamento, mas é o caso mais aplicado na prática, pois nem sempre possuímos "ruas de mão dupla".

O algoritmo para Localização de Facilidades está sendo implementado em Linguagem C++, paralelamente com o estudo desta linguagem de programação, uma vez que esta linguagem não está totalmente assimilada. Pretende-se usar o compilador DJGPP, por usar instruções em Linguagem C++, e por possuir várias bibliotecas muito úteis, além da característica desejável de gerar seus códigos-fonte com portabilidade para ambientes UNIX e Windows.

DEFINIÇÕES E FORMULAÇÃO

Seja $G(V,E)$ um grafo formado por um conjunto de nós (ou vértices) $V = \{v_1, v_2, \ldots, v_n\}$ representados por pontos e um conjunto de arestas $E = \{e_1\ , e_2, \ldots, e_n\}$.

As arestas podem ser direcionadas, ou seja, podem ter um sentido de fluxo, então G é um grafo orientado e suas arestas são chamadas de **arcos**. Os arcos possuem valores atribuídos à eles, então G se torna um grafo ponderado por possuir arcos valorados.

Um **caminho** em um grafo é uma seqüência de nós e arestas (ou arcos) adjacentes formando um percurso de um nó à outro. Um **ciclo** é um caminho onde o nó inicial e o final são coincidentes.

Um Ciclo Euleriano é um percurso em que se passa uma vez por cada arco do grafo, começando e terminando no mesmo nó.

Os algoritmos para Localização de Facilidades, tem como objetivo, a localização de facilidades em um grafo. Dois problemas clássicos de localização de facilidades conhecidos na literatura são o problema dos centros e o problema das medianas. No problema dos centros, o objetivo é minimizar a distância até o ponto mais crítico a ser atendido. No problema das medianas, o objetivo é minimizar a soma da distância de cada um dos vértices à facilidade mais próxima, ponderada por um fator de demanda. A esta localização ótima chama-se de Mediana do grafo.
O que está sendo focalizado atualmente é o problema das medianas.

Problema das Medianas:

Considerando um grafo $G(V,E)$ não orientado com n nós, k um número inteiro positivo não nulo $k = (1, 2, 3...)$, escolhemos k pontos distintos em G sendo indicado como conjunto $X = \{x_1, x_2,... x_{k-1}, x_k\}$. A distância mínima entre quaisquer pontos $x_i \in X$ e o nó j em G será $d(i, j)$. O conjunto S indica o conjunto de nós das medianas localizadas durante a execução do algoritmo e m indica o número de nós do conjunto S.

Algoritmo:

Passo 1: Achar a distância mínima entre todos os pares de nós do grafo G usando o algoritmo de Floyd e montar uma matriz de distâncias mínimas (com n linhas por n colunas);

Passo 2: Multiplicar a j-ésima coluna da matriz de distâncias mínimas pela demanda correspondente da carga, com $j = (1, 2, 3, ..., n)$, para obter uma outra matriz $\lfloor h_j * d(i,j) \rfloor$;

Passo 3: Para cada linha i da matriz $\lfloor h_j * d(i,j) \rfloor$, faça a soma de todos os elementos da linha. O nó correspondente a linha cuja soma de seus elementos for o menor valor obtido é a localização da 1-mediana;

Passo 4: Admite-se que $m = 1$ e que a 1-mediana está em i, então $S = \{i\}$;

Passo 5: Inclui-se uma nova facilidade no conjunto S, escolhendo a localização entre os nós de $N - S$, ou seja, os nós que não estão em S, no qual provém uma maior possibilidade de melhorar a função objetivo com o número das medianas incrementadas em 1. Faça-se $m \leftarrow m + 1$;

Passo 6: Na tentativa de melhorar a função objetivo com a substituição de um dos nós em S com um nó que está em $N - S$. A cada vez que uma solução melhorada for obtida, ela deve ser usada como uma nova solução do conjunto S, então repete-se este passo. Quando todas as possibilidades de substituição de nós do conjunto S forem tentadas sem melhorar a função objetiva, execute o Passo 7;

Passo 7: Se $m = k$, pare; caso contrário, retorne ao Passo 5.

Carteiro Chinês em Grafos orientados

O algoritmo para resolução do Problema do Carteiro Chinês em grafos orientados consiste em verificar se o número de arcos que entram e o número de arcos que saem de cada um dos nós do grafo são iguais, assim possibilitando a obtenção do Ciclo Euleriano. Chama-se P_i a polaridade do nó i e define-se como o número de arcos que entram menos o número de arcos que saem do nó i. Quando P_i for maior que zero, dizemos que i é um nó de oferta, caso P_i for menor que zero, i será um nó de demanda. Os conjuntos do nós de oferta e de demanda são chamados de O e D, respectivamente.

Algoritmo:

Passo 1: Verificar quais os nós de oferta e os de demanda, carregando o P_i de cada um e as distâncias mínimas $d(i,j)$ de todos os nós $i \in O$ para todos os nós $j \in D$.
Passo 2: Faça combinações de nós de oferta com os de demanda, de modo a minimizar $\sum_{i \in O} \sum_{j \in D} d(i,j) x_{ij}$, com $\sum_{j \in D} x_{ij} = P_i$ para todo $i \in O$, $\sum_{i \in O} x_{ij} = -P_j$ para todo $j \in D$ e $x_{ij} \geq 0$.

Passo 3: Para cada $x_{ij} > 0$ encontrado, é adicionado ao grafo G, x_{ij} cópias dos caminhos mínimos de $i \in O$ à $j \in D$, gerando um grafo G' com todos os $P_i = 0$ para todos os nós.
Passo 4: Ache então o Ciclo Euleriano de G'. Este ciclo é a resolução do Problema do Carteiro Chinês para o grafo G.

PRÓXIMAS ETAPAS

A implementação dos algoritmos descritos na seção anterior já foi iniciada. Esta implementação está sendo feita de modo que qualquer usuário, de PC ou Estação, possa entrar com os dados do grafo, manualmente ou através de um arquivo texto, e fazer uso dos algoritmos descritos.

BIBLIOGRAFIA

Boaventura Netto, P. O.	Teoria e Modelos de Grafos, Editora Edgard Blücher, 1979;
Christofides, N.	Graph Theory – An Algorithmic Approach, Editora Academic Press INC., 1975;
Larson, R. C.; Odoni, A. R.	Urban Operations Research, Editora Prentice-Hall, 1981;
Motta, I. V. M.;	Roteamento de Veículos para Distribuição de Água, Publicação INPE 4318 – TDL/285, 1987;
Pappas, C H; Murray, W H	Turbo C++ Completo e Total, Editora McGraw-Hill & MAKRON Books Ltda., 1991;
Sampaio, Rudini M.	Estudo e Implementação de Algoritmos de Roteamento, Trabalho de Graduação CTA/ITA, 1998;
Szwarcfiter, J. L.	Grafos e Algoritmos Computacionais, Editora Campus, 1984.